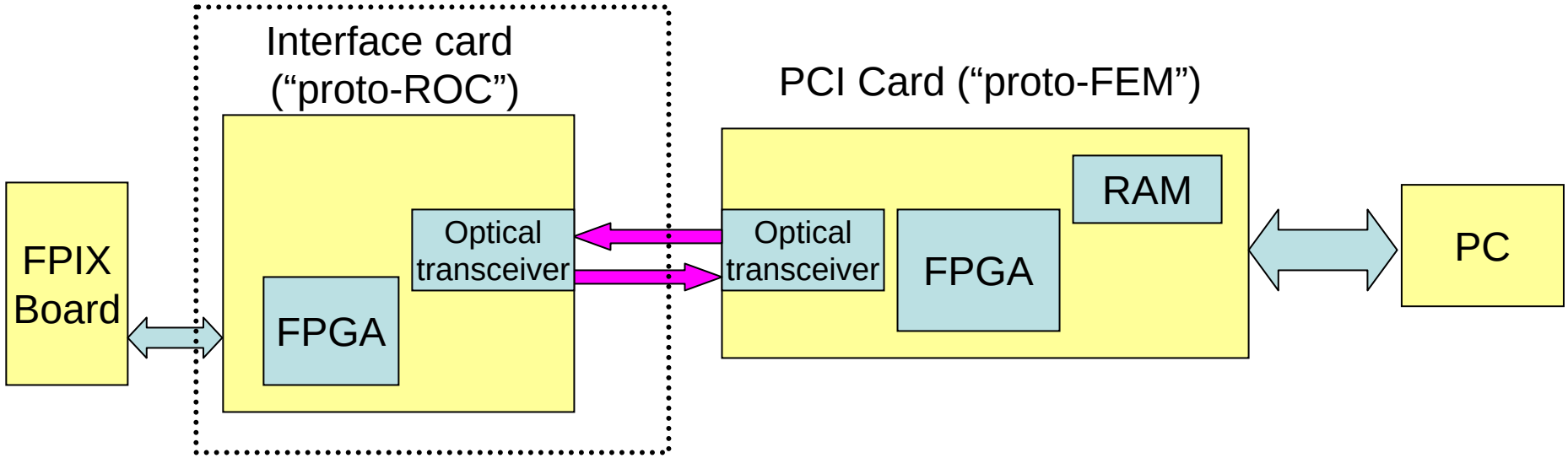
Nevis FVTX Update

Dave Winter

FVTX Silicon Meeting

13 July 2006

Nevis FPIX Test-stand



pROC

- Altera Cyclone
- 2Mb serial eprom
- CYP15G0101DXB serializer
- Agilent HFBR 53A3DEM
- Interfaces with FNAL FPIX test board

pFEM

- PCI-based
- Altera ACEX
- CYP15G0101DXB serializer
- 256k x 36bits SRAM
  - Provides I/O buffers
- Agilent HFBR 53A3DEM

PC

- Linux
- Interactive shell to send commands to all three boards
- Can tx/rx basic FPIX commands

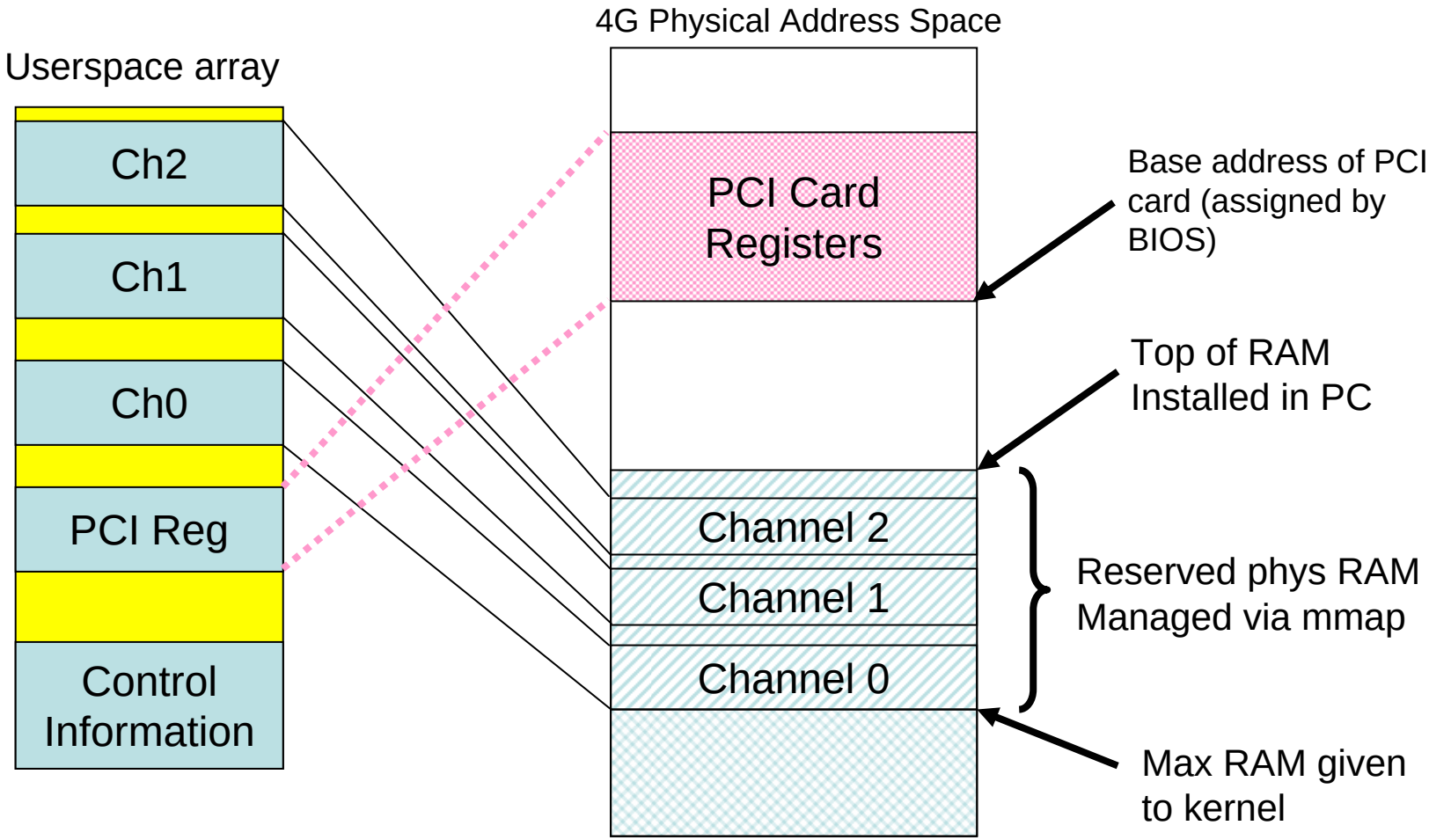
User-PCI Card communication

- Three communication channels defined
  - Channel 0: Control commands to PCI and FVTX
  - Channel 1: Response to Control Commands
  - Channel 2: Data from FPIX
- Channels implemented as ring buffers (Bruce's ABC architecture):
  - A & B are r/w pointers (which is which depends on direction of data)
  - C is word that contains info about how the hardware should wrap ptrs
- Two fiber channels: Transmit & Receive
  - Data to/from fiber buffered in on-board SRAM
- Channel data passed between user and card thru reserved physical PC RAM managed by user process
  - 3 RAM regions + PCI registers mapped to user space
  - PCI card executes DMA transfers to/from RAM
- Packet structure defined to allow PCI and FPIX commands to be passed back and forth

PCI Card Registers

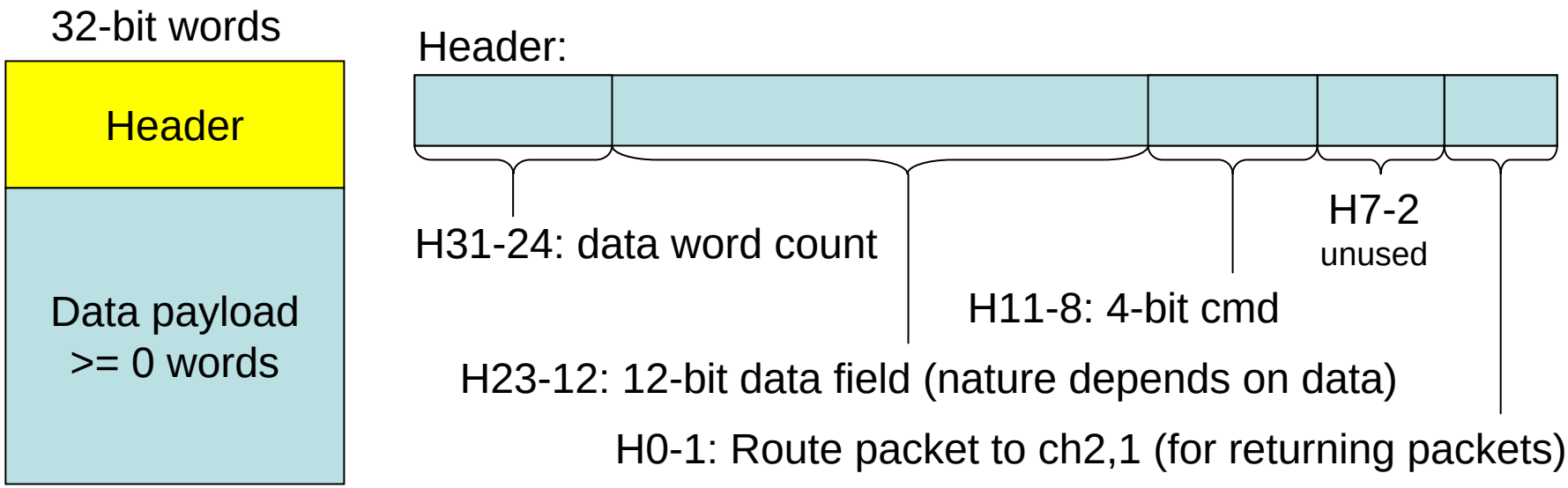
- Standard 32-bit 33 MHz PCI card
- 32 32-bit registers starting at PCI device base address
  - 0: Device and Vendor ID (readonly)
  - 1: Status/Command - most bits nonexistant, but some R/W
    - b0: Status/Command - most bits nonexistant, but some R/W
    - b1: BM Bus Master enable - enables dma activity
  - 2: readonly class code and revision id
  - 4: Base Address - pci address of first register
  - 16: CSR (Control and Status Register)
  - 17, 18, 19: ABC regs for PCI to fiber transfers
  - 20, 21, 22: ABC regs for fiber to PCI port 0
  - 23, 24, 25: ABC regs for fiber to PCI port 1
- CSR (Control and Status Register)
  - CSR[2..0] enable dma activity for each of the 3 channels
  - CSR[5..3] enable fiber activity for each of the 3 channels
  - CSR[6] is a flag to indicate loss of signal on fiber input
  - CSR[9..7] are AEB flags for each of 3 channels
  - CSR[31] = 1 resets PCI Card

User space mapping



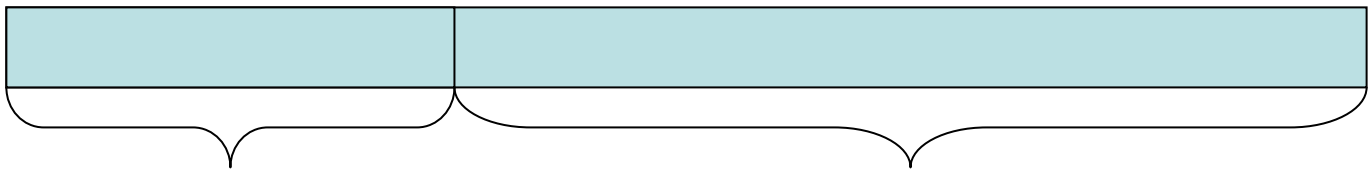
- Accessing any of the registers or the channels' ringbuffers is as easy as accessing the appropriate array element

Packet Structure



- 4-bit Commands (H11-8)
  - 0: normal fpix command with H[23..12]=shift count followed by data words: 13 bits plus data
  - 1: generate firefighter reset for fpix
  - 2: return recent sync word (10-bit status) in H[21..12]
  - 3: test pulse with 6-bit amplitude in H[17..12]
  - 4: force hard reset of fpixtest and fpix
  - 5: read/write CSR of fpixtest card

Packet Structure: Data Words



D31-19: 13-bit FPIX command (optional) Data bits

- Outgoing: typically will be commands to be delivered directly to FPIX
  - For data to be written to FPIX, shift count is contained in data field of header
- Incoming: Echoed FPIX command + result of response to command (where applicable)
- If data is included (for those FPIX commands requiring it), ordering is done according to expectation of FPIX
  - Most-significant bit first for FPIX command
  - Least significant bit first for data bits
  - Could change in the future (ie. let ROC FGPA reorder bits as needed)

PCI Card-Interface Card Communication

- Currently the only path of communication to interface board is fiber
- Takes data (control commands) queued up in SRAM, serializes, and sends over fiber
- Receives data (control commands and FPIX data) from fiber, deserializes and queues up to SRAM
- Packet format can contain either interface or fpix commands, PCI card doesn't care
  - Only cares if a full packet is present to send

Future

- Considering switch to 16-bit packet structure
- Redesign PCI card to include basic inputs required of FEM
- Next generation of interface card to be able to interface to 8-chip module?